CLAIMS

Please amend the claims as follows:

1. (Original) An apparatus for synchronizing an asynchronous divider reset signal for a plurality of non-integer frequency ratio clocking signals, comprising:

a sampling circuit to at least derive a syncClk clocking signal from the plurality of non-integer frequency ratio clocking signals;

a plurality of synchronization latches at least configured to toggle on an edge of the syncClk clocking signal, wherein at least one synchronization latch of the plurality of latches is at least configured to receive the asynchronous divider reset signal, and wherein at least one synchronization latch of the plurality of synchronization latches is at least configured to produce a sync signal; and

a counting circuit at least configured to receive the sync signal and at least configured to output a synchronous divider reset signal.

2. (Original) The apparatus of Claim 1, wherein the plurality of non-integer frequency ratio clocking signals further comprises two synchronous input clocking signals.

3. (Original) The apparatus of Claim 1, wherein the sampling circuit further comprises:

a delay element, wherein the delay element is at least configured to receive a selected sampling clocking signal of the two synchronous input clocking signals and is at least configured to generate a time delayed version of the selected sampling clocking signal; and

a sampling latch, wherein the sampling latch is at least configured to receive a non-sampling clocking signal of the two synchronous input clocking signals and is at least configured to generate a syncClk clocking signal, and wherein the sampling latch is at least configured to trigger on an edge of the time delayed version of the selected sampling clocking signal.

4. (Original) The apparatus of Claim 3, wherein the sampling latch further comprises at least one D flip-flop.

5. (Original) The apparatus of Claim 4, wherein the D flip-flop is at least configured to:

receive a non-sampling clocking signal of the two synchronous input clocking signals and is at least configured to generate a syncClk clocking signal; and

trigger on an edge of the time delayed version of the selected sampling clocking signal.

6. (Original) The apparatus of Claim 1, wherein the plurality of synchronization latches further comprises a plurality of cascaded D flip-flops.

7. (Original) The apparatus of Claim 6, wherein the plurality of cascaded D flip-flops further comprises:

a first D flip-flop, wherein the first D flip-flop is at least configured to receive the asynchronous divider reset signal, and wherein the first D flip-flop is at least configured to trigger on an edge of the syncClk clocking signal; and

a second D flip-flop, wherein the second D flip-flop is at least configured to receive the output of the first D flip-flop and is at least configured to generate the sync signal, and wherein the second D flip-flop is at least configured to trigger on the edge of the syncClk clocking signal that triggers the first D flip-flop.

8. (Original) The apparatus of Claim 1, wherein the counting circuit further comprises:

a plurality of counting latches, wherein at least one counting latch of the plurality of counting latches is at least configured to receive the sync signal; and

an XOR gate, wherein the XOR gate is at least configured to receive an output of at least two counting latches of the plurality of counting latches and is at least configured to output the synchronous divider reset signal.

9. (Original) The apparatus of Claim 8, wherein the plurality of counting latches further comprises a plurality of D flip-flops.

10. (Original) The apparatus of Claim 9, wherein the plurality of non-integer frequency ratio clocking signals further comprises two synchronous input clocking signals.

11. (Currently Amended) The apparatus of Claim 10, wherein the plurality of D flip-flops further comprises:

a first D flip-flop, wherein the first D flip-flop is at least configured to receive the sync signal, and wherein the first D flip-flop is at least configured to trigger on an edge of the selected sampling clocking signal of the two synchronous input clocking signals to output a first counting output; and

a plurality of additional cascaded D flip-flops, wherein a first cascade D flip-flop of the plurality of cascaded D flip-flops is at least configured to receive the first counting output, and wherein each of the cascade D flip-flops of the plurality of cascaded D flip-flops is at least configured to trigger on an edge of the selected sampling clocking signal of the two synchronous input clocking signals.

12. (Original) A method for synchronizing an asynchronous divider reset signal for a plurality of non-integer frequency ratio clocking signals, comprising:

receiving the asynchronous divider reset signal;

producing a syncClk clocking signal from the plurality of non-integer frequency ratio clocking signals;

producing a sync signal from the asynchronous divider reset signal, wherein the sync signal is at least triggered on an edge of the syncClk clocking signal;

generating a plurality of cascaded counting signals from the sync signal, wherein the counting signals are at least triggered on an edge of at least one non-integer frequency ratio clocking signal of the plurality of non-integer frequency ratio clocking signals; and

operating on at least two cascaded counting signals of the plurality of counting signals with a logic gate to produce a synchronous divider reset signal.

13. (Original) The method of Claim 12, wherein the step of producing a syncClk clocking signal further comprises:

delaying at least one non-integer frequency ratio clocking signal of the plurality of non-integer frequency ratio clocking signals to produce a delayed sampling clocking signal;

toggling a sampling latch with at least one non-sampling clocking signal of the plurality of non-integer frequency ratio clocking signals; and

triggering the sampling latch with an edge of the delayed sampling clocking signal.

14. (Original) The method of Claim 13, wherein the step of producing the syncClk clocking signal further comprises receiving two non-integer frequency ratio synchronous clocking signals.

15. (Original) The method of Claim 12, wherein the step of producing a sync signal further comprises:

toggling a first synchronization latch with the asynchronous divider reset signal to produce a first output;

triggering the first synchronization latch with an edge of the syncClk clocking signal;

toggling a second synchronization latch with the first output to produce the sync signal; and

triggering the second synchronization latch with the edge of the syncClk clocking signal that triggers the first synchronization latch.

16. (Original) The method of Claim 12, wherein the step of generating a plurality of cascaded counting signals further comprises:

toggling a first counting latch with the sync signal to produce a first counting output;

propagating the first counting output with an edge of a sampling clocking signal of the plurality of non-integer frequency ratio clocking signals to produce a first cascade signal;

serially toggling a plurality of serially coupled latches to produce a plurality of cascaded counting signals; and

triggering each of the cascaded counting signals of the plurality of cascaded counting signals with an edge of a sampling clocking signal of the plurality of non-integer frequency ratio clocking signals.

17. (Original) The method of Claim 12, wherein the step of operating further comprises XORing two cascaded counting signals to produce the synchronous divider reset signal.

18. (Original) A computer program product for synchronizing an asynchronous divider reset signal for a plurality of non-integer frequency ratio clocks, with the computer program product

having a medium with a computer program embodied thereon, wherein the computer program comprises:

computer code for receiving the asynchronous divider reset signal;

computer code for producing a syncClk clocking signal from the plurality of non-integer frequency ratio clocking signals;

computer code for producing a sync signal from the asynchronous divider reset signal, wherein the sync signal is at least triggered on an edge of the syncClk clocking signal;

computer code for generating a plurality of cascaded counting signals from the sync signal, wherein the counting signals are at least triggered on an edge of at least one non-integer frequency ratio clock of the plurality of non-integer frequency ratio clocking signals; and

computer code for operating on at least two cascaded counting signals of the plurality of counting signals with a logic gate to produce a synchronous divider reset signal.

19. (Original) The computer program product of Claim 18, wherein the computer code for producing a syncClk clocking signal further comprises:

computer code for delaying at least one non-integer frequency ratio clocking signal of the plurality of non-integer frequency ratio clocking signals to produce a delayed sampling clocking signal;

computer code for toggling a sampling latch with at least one non-sampling clocking signal of the plurality of non-integer frequency ratio clocking signals; and

computer code for triggering the sampling latch with an edge of the delayed sampling clocking signal.

20. (Original) The computer program product of Claim 19, wherein the computer code for producing the syncClk clocking signal further comprises receiving two non-integer frequency ratio synchronous clocking signals.

21. (Original) The computer program product of Claim 18, wherein the computer code for producing a sync signal further comprises:

computer code for toggling a first synchronization latch with the asynchronous divider reset signal to produce a first output;

computer code for triggering the first synchronization latch with an edge of the syncClk clocking signal;

computer code for toggling a second synchronization latch with the first output to produce the sync signal; and

computer code for triggering the second synchronization latch with the edge of the syncClk clocking signal that triggers the first synchronization latch.

22. (Original) The computer program product of Claim 18, wherein the computer code for generating a plurality of cascaded counting signals further comprises:

computer code for toggling a first counting latch with the sync signal to produce a first counting output;

computer code for propagating the first counting output with an edge of a sampling clocking signal of the plurality of non-integer frequency ratio clocking signals to produce a first cascade signal;

computer code for serially toggling a plurality of serially coupled latches to produce a plurality of cascaded counting signals; and

computer code for triggering each of the cascaded counting signals of the plurality of cascaded counting signals with an edge of a sampling clocking signal of the plurality of non-integer frequency ratio clocking signals.

23. (Original) The computer program product of Claim 18, wherein the computer code for operating further comprises XORing two cascaded counting signals to produce the synchronous divider reset signal.

24. (Cancelled) An apparatus for ensuring synchronization of clocks in a multiple clock processor system having a plurality of downstream divider stream trees, comprising:

first logic means for indicating misalignment of first and second clocks;

reset means, connected to said first logic means, for providing a synchronous divider reset signal for resetting each of said downstream divider trees; and

second logic means, connected to first logic means and said reset means, for receiving said first logic means indication and for determining a proper clock cycle at which to issue a signal to the

reset means, wherein the reset means is configured for providing the synchronous divider reset signal in response to the received signal from the second logic means.

25. (Currently Amended) ~~The apparatus of Claim 24, wherein the first logic means further comprises:~~ An apparatus for ensuring synchronization of clocks in a multiple clock processor system having a plurality of downstream divider stream trees, comprising:

first logic means for indicating misalignment of first and second clocks;

reset means, connected to said first logic means, for providing a synchronous divider reset signal for resetting each of said downstream divider trees;

second logic means, connected to first logic means and said reset means, for receiving said first logic means indication and for determining a proper clock cycle at which to issue a signal to the reset means, wherein the reset means is configured for providing the synchronous divider reset signal in response to the received signal from the second logic means; and

wherein the first logic means further comprises:

a delay element, wherein the delay element is at least configured to receive at least one of the clocking signals selected from the group consisting of said first and second clocks and is at least configured to generate a time delayed version of the selected sampling clocking signal; and

a sampling latch, wherein the sampling latch is at least configured to receive an output of the delay element.

26. (Currently Amended) The apparatus of Claim 25[[24]], wherein the reset means further comprises a plurality of synchronization latches, wherein at least one latch of the plurality of synchronization latches is at least configured to receive an[[the]] asynchronous divider reset signal, and wherein at least one synchronization latch of the plurality of synchronization latches is at least configured to produce a sync signal.

27. (Previously Presented) The apparatus of Claim 26, wherein the plurality of latches further comprises a plurality of cascaded D flip-flops.

28. (Currently Amended) The apparatus of Claim 25[[24]], wherein the second logic further comprises a counting circuit.

29. (Currently Amended) ~~The apparatus of Claim 28, wherein the counting circuit further comprises:~~ An apparatus for ensuring synchronization of clocks in a multiple clock processor system having a plurality of downstream divider stream trees, comprising:

first logic means for indicating misalignment of first and second clocks;

reset means, connected to said first logic means, for providing a synchronous divider reset signal for resetting each of said downstream divider trees;

second logic means, connected to first logic means and said reset means, for receiving said first logic means indication and for determining a proper clock cycle at which to issue a signal to the reset means, wherein the reset means is configured for providing the synchronous divider reset signal in response to the received signal from the second logic means; and

wherein the second logic further comprises a counting circuit, comprising:

a plurality of counting latches, wherein at least one counting latch of the plurality of counting latches is at least configured to receive a signal from the reset means; and

an XOR gate, wherein the XOR gate is at least configured to receive an output of at least two counting latches of the plurality of counting latches and is at least configured to output the synchronous divider reset signal.